VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Vytvoření a analýza sociální sítě nad vybranými zdroji na Webu.

The construction and analysis of social network on the Web.

2010 Petr Dolenský

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

V Ostravě 3. května 2010 .....

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 3. května 2010 .....

Tímto bych chtěl poděkovat všem, kteří mi s tvorbou této práce pomohli, zvláště pak Mgr. Pavle Dráždilové za její odbornou asistenci a vedení této bakalářské práce.

Abstrakt

Náplní bakalářské práce je automatizovaná extrakce a zpracování vybraných dat na webu, konkrétně diskuzních fór. Uložení extrahovaných dat do datových struktur vhodných pro další zpracovávání. Pokus o vytvoření sociální sítě nad získanými daty. Následná analýza, interpretace a vizualizace získaných výsledků.

Klíčová slova: sociální síť, analýza sociální sítě, centralita, diskuzní fórum, C♯, regulární výrazy

Abstract

The author deals with automatic extraction and processing of selected data from the web. Storing the exctracted data in suitable data structures. The author tries to create a social network above the selected data, followed by further analysis, interpretation and visualition of the obtained results.

Keywords: social network, social network analysis, centrality, internet forum, C♯, regular expressions

Seznam použitých zkratek a symbolů

HTML	–	Hyper Text Markup Language
HTTP	–	Hyper Text Transfer Protocol
URL	–	Uniform Resource Locator
SQL	–	Structured Query Language
ERD	–	Entity-relationship Diagram
IRC	–	Internet Relay Chat
UTF	–	UCS Transformation Format
ID	–	IDentification

Obsah

Seznam tabulek

Seznam obrázků

Seznam výpisů zdrojového kódu

1 Úvod

Obsahem bakalářské práce je extrakce a analýza dat na webu. Jako zdroj dat jsem zvolil dvě aktivní diskuzní fóra.

Diskuzní fóra jsem si vybral, protože mají dnes na internetu svoji nezastupitelnou roli. Pro mnoho lidí je návštěva a účast v diskuzích na internetu součástí každodenní rutiny. Diskuzní fóra sdružují přátele, lidi se společnými zájmy, vírou, lidi účastnící se stejných akcí, poslouchající stejnou hudbu atd. Toto platí i pro sociální sítě, ale vazby mezi jednotlivými účastníky diskuzí nemusí být na první pohled tak zjevné. Někteří lidé mohou stejné fórum navštěvovat pravidelně několik let, naopak někdo může navštívit tématicky zaměřené fórum jen jednou, aby získal radu, doporučení či se jen zeptal zkušenějších uživatelů v daném oboru, a dále se diskuzí už neúčastní. Takto velké a rozmanité prostředí je vhodným subjektem k podrobnému zkoumání a analýze.

Pokud chceme nad diskuzním fórem vytvořit socialní síť, je potřeba z diskuzního fóra extrahovat informace, které jsou nezbytné pro nalezení vazeb a relací mezi uživateli, které tvoří případnou sociální síť.

Obě mnou zvolená diskuzní fóra jsou rozdílná strukturou, tématicky, i počtem uživatelů. První fórum sdružuje uživatele z celého světa a jazykem je angličtina, druhé sdružuje převážně české uživatele a hovoří se na něm výhradně česky. Obě fóra jsou relativně úzce zaměřená svým tématickým obsahem (pěstování rostlin, silové sporty).

V první části práce stručně popíši obsah a strukturu těchto fór. Následuje kapitola věnovaná automatizované extrakci dat, nezbytné pro další zpracovávání. Další kapitoly se zabývají analýzou a pokusy nad získanými daty, interpretací a vizualizací výsledků.

Jako programovací jazyk pro aklikaci extrahující data z diskuzních fór jsem zvolil **C♯** na platformě **.NET**. Pro ukládání dat jsem použil databázový systém **MySQL** a MySQL server 5.0.

2 Diskuzní fóra

V této kapitole podrobněji popíšu zvolená diskuzní fóra. Pro efektivní extrakci dat je nezbytná správná analýza struktury fóra, provázanost jednotlivých témat, příspěvků a účastníků.

Internetové diskuzní fórum je místo na internetu, kam lidé vkládají svoje názory nebo reakce, a ty se následně na stránce zobrazují. Dnešní diskuzní fórum je moderním ekvivalentem tradiční nástěnky. Z technického hlediska jsou diskuzní fóra webové aplikace, které spravují výhradně uživatelem generovaný obsah.

Diskuzní fórum umožňuje lidem na internetu účastnit se již probíhajících diskuzí, nebo zakládat nové. Fórum je zpravidla hierarchicky členěno do jednotlivých tématických sekcí. Jednotlivé diskuze bývají vymezeny diskuzním tématem.

Oproti IRC kanálům nebo chatu se internetová diskuze obvykle liší tím, že přispěvatelé nemusí být ke stránce připojeni současně a reagovat bezprostředně, ale mohou reagovat i s odstupem mnoha dní či měsíců (viz [12]). Témata jsou tvořena příspěvky uživatelů, které se za sebou řadí v chronologickém pořadí.

Účast v diskuzích a zakládání nových témat je zpravidla podmíněno registrací uživatelů na konkrétním diskuzním serveru. Registrace uživatelů zaručuje, že pod určitým jménem nebo přezdívkou nebude v diskuzi vystupovat nikdo jiný, než ten, kdo si ji zaregistroval. Podmínkou registrace někdy může být uvedení osobních údajů, uhrazení účastnického poplatku nebo splnění jiných podmínek stanovených provozovatelem nebo moderátorem (viz [12]). Jelikož nelze neregistrovaného uživatele fóra jednoznačně identifikovat, rozhodl jsem se takové uživatele a příspěvky ignorovat, a vyloučit z extrakce i analýzy.

2.1 Diskuzní fórum http://www.cpukforum.com

Diskuzní fórum **http://www.cpukforum.com** (dále jen cpukforum.com) je anglické fórum sdružující pěstitele masožravých rostlin po celém světě. Fórum jako takové běží od roku 2002 a v současné době čítá okolo 150 000 příspěvků a více než 3500 registrovaných uživatelů.

Fórum **cpukforum.com** je přirozeně rozděleno do jednotlivých tématických sekcí (láčkovky, rosnatky, pěstování ve skleníku, pěstování v teráriu, nákup a prodej atd.). Jednotlivé sekce mohou obsahovat více podsekcí, nebo přímo samotná témata (anglicky *topics*). Každé téma je tvořeno posloupností jednoho nebo více příspěvků zaslaných uživateli.

2.2 Diskuzní fórum http://www.ironfactor.cz

Diskuzní fórum **http://www.ironfactor.cz** je české fórum o silových sportech. V současné době čítá fórum okolo 110 000 příspěvků a téměř 1300 registrovaných uživatelů. Toto fórum je stejně jako první fórum rozděleno do tématických sekcí (trénink, výživa, soutěže atd.), které už obsahují samotná témata.

3 Extrakce dat

V této kapitole rozeberu, jakým způsobem jsem data z diskuzních fór extrahoval.

Nejprve jsem v jazyce C♯ naprogramoval aplikaci, která prochází webové stránky diskuzních fór a ukládá celý jejich HTML kód. Tento způsob získávání informací se nazývá Web Scraping (viz 3.0.1). HTML kód je získán jako řetězec znaků (String). Následně z kódu extrahuji pouze data, která mě zajímají.

Program se v kódu orientuje dle HTML tagů, které definují strukturu a výslednou podobu stránky ve webovém prohlížači. Za pomoci zejména regulárních výrazů je možné extrahovat např. titulek webové stránky, všechny URL odkazy jež stránka obsahuje, bloky HTML kódu jenž vymezují jednotlivé příspěvky v diskuzních fórech. Tyto bloky jsou následně programem opět rozděleny (extrakce informací o uživateli, datumy atd.) a ukládány do MySQL databáze.

Abysme takto mohli projít a analyzovat celé diskuzní fórum, je třeba předem důkladně prostudovat jeho HTML kód. Každé diskuzní fórum je jiné a z toho vyplýva, že program je potřřeba napsat tzv. "na míru" každému diskuznímu fóru. Blíže se na daný problém podívame v kapitole 3.1.

Následuje ukázka zdrojového kódu programu pro extrakci dat, konkrétně získání HTML kódu z webové stránky diskuzního fóra.

```
{

  WebClient webClient = new WebClient();
  const string strUrl  = "http :// www.cpukforum.com/forum/index.php?";
  byte[] reqHTML;
  reqHTML = webClient.DownloadData(strUrl);
  UTF8Encoding objUTF8 = new UTF8Encoding();
  lblWebpage.Text = objUTF8.GetString(reqHTML);

}
```

Výpis 1: Extrakce HTML kódu z webové stránky.

3.0.1 Web scraping

Web scraping je proces automatického skenování HTML kódu webových stránek, zpravidla vykonávaný programem, který postupně prochází a ukládá HTML kódy webových stránek. Tato metoda je hojně využívána programy (roboty), jejichž úkolem je simulovat chování člověka na webu a sbírat velká množství dat.

3.1 Extrakce dat z diskuzního fóra

Jak jsem již zmínil výše, nejprve bylo třeba napsat program, který hierarchicky prochází fórum v následujícím pořadí:

1. Sekce
2. Téma (*topic*)
3. Příspěvek
4. Informace o uživateli

Program prochází úvodní stránku fóra a ukládá všechny URL odkazy (dále jen odkazy) podsekcí, které mají v tomto případě následující tvar:

http://www.cpukforum.com/forum/index.php?showforum=1

http://www.ironfactor.cz/viewforum.php?f=2

Číslo na konci odkazu je unikátní pro každou sekci a dobře poslouží jako identifikátor (primární klíč) jednotlivých sekcí v databázi. Což je rozhodně vhodnější, než si pamatovat dlouhý název podsekce.

Následně procházím všechny nasbírané odkazy podsekcí a ukládám všechny odkazy na jednotlivá témata ve tvaru:

http://www.cpukforum.com/forum/index.php?showtopic=34636

http://www.ironfactor.cz/viewtopic.php?t=3526

Číslo na konci odkazu je unikátní pro každé téma a slouží jako jedinečný identifikátor (primární klíč) tématu v databázi.

V HTML kódu jednotlivých témat v cyklu hledám HTML tagy vymezující příspěvky uživatelů. Ty dále rozděluji, a extrahuji potřebné informace:

1. Číslo příspěvku (unikátní)
2. Pořadí příspěvku (v tématu)
3. Text příspěvku
4. Datum vložení příspěvku

Dále extrahuji informace o uživateli, který příspěvek zaslal:

1. Číslo uživatele (unikátní)
2. Jméno
3. Další údaje (celkový počet příspěvků, odkud je, atd.)

Pozornost je třeba věnovat vícestránkovým podsekcím a tématům. Webová stránka fóra většinou umožnuje zobrazit pouze omezený počet témat v sekci (příspěvků v tématu) na jednu stránku, zpravidla 15 až 30. V HTML kódu každé sekce a tématu je potřeba rozpoznat, zda se neskládá z více stránek. Pokud naleznu blok HTML kódu, který ve webovém prohlížeči nabízí přechody mezi stránkami, v závislosti na počtu stránek získaného z HTML kódu vygeneruji odkazy všech stránek jednotlivých sekcí. Přesný zápis odkazu pro přístup k jednotlivých stránkám sekcí či témat závisí na počtu témat/příspěvků zobrazených na jedné stránce. Takto získané odkazy dále zpracovávám.

3.2 Ukládání dat

Extrahovaná data je potřeba ukládat do vhodných datavých struktur pro pozdější zpracování a analýzu.

Při psaní programu pro extrakci dat jsem zpracovaná data ukládal pro rychlou kontrolu rovnou na pevný disk. Každá podsekce tvořila samostatný adresář a každé téma představovalo jeden textový soubor. Název textového souboru tvořilo unikátní číslo tématu (odpovídající číslu tématu diskuzního fóra na webu) a název tématu. Obsahem textového souboru byla posloupnost jednotlivých příspěvků. Každé tělo příspěvku uvozovala hlavička, ve které byly všechny extrahované informace týkající se daného příspěvku (číslo, datum, atd.) a uživatele, který příspěvek zaslal (jméno, adresa, celkový počet příspěvků, skupina, atd.). Takto vytvořená struktura byla v měnším měřítku dobře přehledná a posloužila jako rychlá zpětná vazba zda program pracuje správně a extrahuje data korektně.

Výše popsaná strukutra však není vhodná pro rychlé zpracovávání většího objemu dat. S řádově desítkami tisíc příspěvků a stovkami uživatelů by se obtížně manipulovalo. Vzhledem k tomu, že původní diskuzní fóra jsou na serveru, na kterém běží, bezpochyby uložena v databázích, použil jsem jako datovou strukturu také databázi. Zvolil jsem databázový systém **MySQL**. Pro komunikaci programu s databází je nezbytná spuštěná instance **MySQL server 5.0** a knihovna **Connector/NET** pro propojení a komunikaci **.NET** a **MySQL**.

Práce s databází značně usnadňuje manipulaci s daty. Údaje lze velmi rychle třídit, vyhledávat atd.

Po vytvoření databáze program komunikuje s databází zasíláním SQL příkazů na MySQL server běžící např. na lokálním počítači (localhost). Namísto vytváření textových souborů jsou do databáze vkládány jednotlivé záznamy (sekce, témata, příspěvky a uživatelé) s odpovídajícími atributy.

Následující ukázka SQL příkazu demonstruje vložení 3 záznamů (řádků) do tabulky Uživatel.

```
INSERT INTO `user` (`id_user`,`name`,`group`,`from`,`sig`,`joined`,`rating`,`posts`) VALUES
(22,'Alexis',' Moderator','Manchester','','7th April 2002',1,3356),
(33,'andycpuk','Admin','Midlands − UK','','27th March 2002',1,1519),
(73,'BobZ','Members','northwestern California USA','','3rd January 2003',1,781);
```

Výpis 2: SQL příkaz pro vložení záznamu do databáze

Je třeba zmínit, že do databáze jsou ukládani pouze uživatelé, kteří napsali nejméně jeden příspěvek, nebo založili nejméně jedno téma. To znamená, že výsledný počet uživatelů v databázi neodpovídá počtu registrovaných uživatelů na jednotlivých fórech, jelikož registrovaný uživatel nemusel být na fóru do doby extrakce jakkoliv aktivní.

3.2.1 Práce s datem

Při vkládání datumů do databáze bylo třeba převést anglický textový zápis data na datový typ DATETIME, který v MySQL slouží pro ukládání data a času. Formát MySQL DATETIME má tvar YYYY-MM-DD HH:MM:SS (rok:měsíc:den hodina:minuta:sekunda). Povolený rozsah je od ”1000-01-01 00:00:00” do ”9999-12-31 23:59:59”. Bylo potřeba napsat funkci, která převede např. ”3rd March 2010 - 08:33 AM” do formátu DATETIME: ”2010-03-03 08:33:00”. Někdy je na fóru místo dne uvedeno ”Today” (dnes), případně ”Yesterday” (včera). V takových případech je potřeba brát rok, měsíc a den ze systémového času.

Stejný převod bylo třeba vyřešit v případě českého fóra. S tím rozdílem, že měsíc je psán česky (leden, únor, březen).

3.3 Návrh databáze

Nejprve jsem navrhl a vytvořil databázi v MySQL. Jelikož potřebuji evidovat sekce, témata, příspěvky a uživatele, vytvořil jsem schéma o 4 tabulkách.

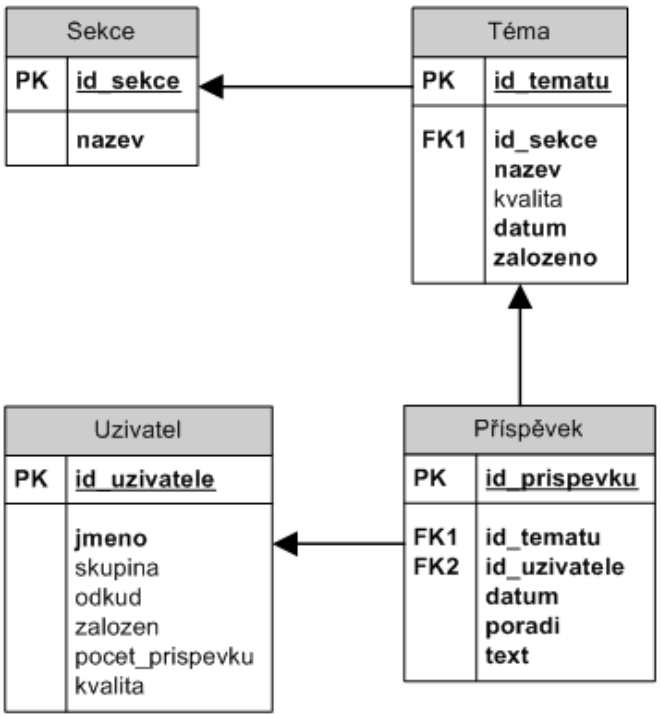
Ačkoliv se programy pro extrakci dat z obou fór liší, zvolil jsem identickou strukturu databáze pro obě diskuzní fóra. Což je velice výhodné pro pozdější dotazování nad databázemi. Jednotlivé SQL dotazy jsou identické, přestože pracují nad různými daty a vracejí rozdílné hodnoty.

3.3.1 Lineární zápis entit a jejich atributů

Lineární zápis entit a atributů popisuje jednotlivé typy entit (tabulky) a jejich atributy jenž evidujeme.

NÁZEV_ENTITY (**Primární klíč**, *Cizí klíč*)

SEKCE (**id_sekce**, název)
TÉMA (**id_tématu**, *id_sekce*, název, kvalita, datum, založeno)
PŘÍSPĚVEK (**id_příspěvku**, *id_tématu*, *id_uživatele*, datum, číslo, text)
UŽIVATEL (**id_uživatele**, jméno, odkud, skupina, počet_příspěvků, kvalita, založen)



Obrázek 1: ER Diagram

3.3.2 ER diagram

ER Diagram na obrázku (1) znázorňuje entity a vztahy mezi nimi. Platí, že:

1. Příspěvek musí patřit právě jednomu uživateli a tématu.
2. Téma musí patřit právě jedné sekci a jednomu uživateli.

3.3.3 Datový slovník

Tabulka (1) reprezentuje datový slovník databáze pro uchování dat z diskuzních fór. Datový slovník zahrnuje seznam všech datových objektů v databázi, jména a popis všech datových prvků a jejich vztahů a údaje o integritních omezeních.

3.3.4 SQL skript pro vytvoření struktury databáze

SQL skript (7) slouží k vytvoření struktury databáze diskuzního fóra **cpukforum.com**. Skript vytvoří 4 tabulky (sekce, téma, příspěvek, uživatel) s odpovídajícími atributy a cizími klíči.

Název Atributu	Datový Typ (velikost)	NULL	Klíč	Popis Atributu
Tabulka Sekce				
id_section	INT(10)	N	PK	Primární klíč sekce
name	VARCHAR(100)	N	N	Název sekce
Tabulka Téma				
id_topic	INT(10)	N	PK	Primární klíč tématu
name	VARCHAR(300)	N	N	Název tématu
id_section	INT(10)	N	FK	Cizí klíč sekce
rating	INT(10)	Y	PK	Kvalita tématu (1-5)
started	DATETIME	N	PK	Datum založení tématu
id_user	INT(10)	N	FK	Cizí klíč uživatele
Tabulka Příspěvek				
id_post	INT(10)	N	PK	Primární klíč příspěvku
id_topic	INT(10)	N	FK	Cizí klíč tématu
id_user	INT(10)	N	FK	Cizí klíč uživatele
date	DATETIME	N	N	Datum zaslání příspěvku
number	INT(10)	N	N	Pořadí příspěvku v tématu
text	VARCHAR(10000)	N	N	Text příspěvku
Tabulka Uživatel				
id_user	INT(10)	N	PK	Primární klíč uživatele
name	VARCHAR(100)	N	N	Jméno uživatele
group	VARCHAR(5)	Y	N	Skupina
from	VARCHAR(100)	Y	N	Odkud je
joined	DATETIME	Y	N	Datum registrace uživatele
rating	INT(10)	Y	N	Hodnocení uživatele (1-5)
posts	INT(10)	Y	N	Celkový počet příspěvků uživatele

Tabulka 1: Datový slovník

```
CREATE DATABASE ‘cpuk‘;

CREATE TABLE ‘cpuk‘.‘post‘ (
  ‘id_post‘ int(10) unsigned NOT NULL,
  ‘id_topic‘ int(10) unsigned NOT NULL,
  ‘id_user‘ int(10) unsigned NOT NULL,
  ‘date‘ datetime NOT NULL,
  ‘number‘ int(10) unsigned NOT NULL,
  ‘text‘ varchar(10000) NOT NULL,
  PRIMARY KEY (‘id_post‘),
  KEY ‘FK_topic‘ (‘id_topic‘) ,
  KEY ‘FK_user‘ (‘id_user‘) ,
  CONSTRAINT ‘FK_topic‘ FOREIGN KEY (‘id_topic‘) REFERENCES ‘topic‘ (‘id_topic‘),
  CONSTRAINT ‘FK_user‘ FOREIGN KEY (‘id_user‘) REFERENCES ‘user‘ (‘id_user‘)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE ‘cpuk‘.‘section‘ (
  ‘id_section‘ int(10) unsigned NOT NULL,
  ‘name‘ varchar(100) NOT NULL,
  PRIMARY KEY USING BTREE (‘id_section‘)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE ‘cpuk‘.‘topic‘ (
  ‘id_topic‘ int(10) unsigned NOT NULL,
  ‘name‘ varchar(300) NOT NULL,
  ‘id_section‘ int(10) unsigned NOT NULL,
  ‘rating‘ int(10) unsigned default NULL,
  ‘started‘ datetime NOT NULL,
  ‘id_user‘ int(10) unsigned NOT NULL,
  PRIMARY KEY (‘id_topic‘),
  KEY ‘FK_section‘ (‘id_section‘) ,
  KEY ‘FK_startedby‘ (‘id_user‘) ,
  CONSTRAINT ‘FK_section‘ FOREIGN KEY (‘id_section‘) REFERENCES ‘section‘ (‘id_section‘),
  CONSTRAINT ‘FK_startedby‘ FOREIGN KEY (‘id_user‘) REFERENCES ‘user‘ (‘id_user‘)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE ‘cpuk‘.‘user‘ (
  ‘id_user‘ int(10) unsigned NOT NULL,
  ‘name‘ varchar(100) NOT NULL,
  ‘group‘ varchar(50) default NULL,
  ‘from‘ varchar(200) default NULL,
  ‘joined‘ varchar(50) default NULL,
  ‘rating‘ int(10) unsigned default NULL,
  ‘posts‘ int(10) unsigned default NULL,
  PRIMARY KEY (‘id_user‘)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Výpis 3: SQL skript pro vytvoření struktury databáze (fórum cpukforum.com).

3.4 Automatizace a použití vláken

Program pro extrakci dat jsem napsal tak, že po spuštění automaticky prochází celé diskuzní fórum, nebo zvolenou oblast. Program extrahovaná data zpracovává, formátuje vhodným způsobem a ukládá do databáze. Celý výše popsaný proces je automatizovaný.

Nicméně postupné procházení webových stránek, jejichž počet se pohybuje v řádech tisíců, je v tomto případě časově náročné. Zpracování jedné podsekce čítající přibližně 1000 témat trvalo programu více než 3 hodiny. Jednoznačně nejvíce času zabere fáze připojení se na HTTP server a extrakce HTML kódu oproti práci s řetězci a databází.

Implementací vláken jsem několikanásobně redukoval čas potřebný pro extrakci a zpracování dat. Program jsem rozdělil na dve části, z nichž každá běží v samostatném vlákně. Vlákna běži paralelně a nezávisle na sobě.

První vlákno se připojuje na HTTP server diskuzního fóra, extrahuje HTML kód stránek a vkládá bloky HTML kódu do sdílené fronty. Druhá část programu (druhé vlákno) vybírá z fronty bloky HTML kódu, zpracovává je požadovaným způsobem a ukládá do databáze).

I při práci s vlákny jsem se přesvědčil, že zpracování a ukládání dat do databáze je rychlejší než extrakce HTML kódu. Vlákno, které má na starost zpracování a ukládání, většinu času pouze čekalo až první vlákno vloží do fronty další HTML stránku určenou ke zpracování.

Následně jsem přidával další vlákna pro extrakci HTML kódu. Zdrojový kód těchto vláken je identický (extrahuj HTML kód, ulož do fronty, opakuj). Vlákna běží paralelně a zpracovává se tak několik různých webových stránek současně.

Výsledkem je program s několika vlákny pro extrakci dat a jedním vláknem (více není potřeba) pro zpracovávání a vkládání do databáze. Celý proces se tak několikanásobně urychlí.

Část programu, kde vlákna přistupují do sdílené paměti (fronty), se nazývá kritická sekce. Pro správný běh programu je třeba zaručit, že žádná dvě vlákna nebudou zapisovat/číst z fronty ve stejný okamžik. Takovýto nežádoucí souběh by mohl vyústit chybnými zápisy dat či pádem programu. C♯ nabízí možnost ošetřit taková místa v programu pomocí funkce Lock(objekt) {}. Funkce zaručuje exkluzivní přístup k objektu (kód ve složených závorkách) v daný okamžik pouze jednomu vláknu.
Ukázka manipulace se sdílenou frontou ve zdrojovém kódu.

```
{
  Lock (sharedQueue)
  {
    sharedQueue.Enqueue(htmlContent);
  }
}
```

Výpis 4: Ošetření kritické sekce pomocí Lock().

3.5 Rozdíly v extrakci dat jednotlivých diskuzních fór

Jak jsem již zmínil na začátku kapitoly Extrakce dat (3), obě diskuzní fóra jsou odlišná zejména strukturou HTML kódu. Bylo tedy třeba napsat program pro extrakci dat pro každé fórum zvlášť.

Ačkoliv je členění obou fór podobné (sekce, téma, příspěvek), obě fóra se liší jak URL odkazy, HTML kódem jednotlivých stránek, jazykem či zobrazovaným formátem času. Jelikož je většina informací z HTML kódu stránek získána pomocí regulárních výrazů, liší se programy právě v použití reg. výrazů.

U každého fóra zvášť, bylo třeba najít místa v HTML kódu, která vymezují jednotlivé příspěvky v rámci tématu a napsat odpovídající regulární výraz. Stejné rozdíly jsou i v reg. výrazech extrahující jména uživatelů, text příspěvku, datum atd.

U obou diskuzních fór jsem také použil rozdílné znakové sady. Pro anglické fórum **cpukforum.com** jsem použil kódování *UTF-8*, nicméně pro české fórum **ironfactor.cz** bylo potřeba použít kódování *windows-1250* pro korektní zobrazování a práci s diakritikou. Rozdílný přístup vyžadoval také zobrazovaný formát data na jednotlivých fórech (viz 3.2.1).

4 Analýza a experimenty s daty

V této kapitole jsem se zabýval analýzou a experimenty nad již získanými daty. V předchozí kapitole (3) je popsán postup, jak jsem data z jednotlivých diskuzních fór získával. V této kapitole popíšu, jak jsem takto získaná data dále zpracovával a analyzoval.

Vstupními daty jsou pro mne dvě stejně strukturované databáze dvou diskuzních fór naplněné daty a informacemi popisující stav fóra, vlastnosti a vazby uživatelů v době extrakce. Jelikož jsou data obou diskuzních fór uložená v MySQL databázi, používám pro získání a porovnávání informací výhradně SQL dotazy typu SELECT * FROM. Veškerá data je možné pohodlně třídit, řadit a seskupovat dle zadaných kritérií. Uživatele je možné seskupovat dle společných atributů, nebo řadit dle počtu založených témat či zaslaných příspěvků.

Zejména pro práci s datem nabízí SQL funkce, které vyrázně usnadňují zisk požadovaných hodnot.

4.1 Statistické údaje

Na data uložená v obou databázích je možné aplikovat různé pohledy a získat zajímavé statistické informace. Je možné porovnávat aktivitu uživatelů v různých časových intervalech, počty různě aktivních uživatelů atd. U obou diskuzních fór jsem porovnával:

1. Aktivitu (počet zaslaných příspěvků) během dne.
2. Aktivitu v rámci týdne.
3. Aktivitu během celého roku.
4. Procento uživatelů, kteří založili téma.
5. Podíl uživatelů s určitým počtem příspěvků.

Takto získané statistické údáje mohou odrážet společné rysy nebo rozdíly obou skupin uživatelů na jednotlivých diskuzních fórech. Blíže takto získané údaje rozeberu v kapitole Interpretace a vizualizace výsledků (5).

4.2 Sociální síť

Definice 4.1 *Sociální síť je společenská struktura tvořená uzly, které jsou obecně jednotlivci, skupiny, nebo organizace. Uzly jsou propojeny různými typy vazeb-závislostmi (zájmy, rodinné vazby, přátelství, víra, návštěvy akcí, diskuze nad společnými tématy, řešení problémů...). Viz [9].*

Termín sociální síť poprvé použil profesor J. A. Barnes v 60. letech, který definoval velikost sociální sítě jako skupinu o počtu 100 až 150 lidí. Barnes studoval vztahy mezi lidmi žijícími v osadě na norských ostrovech (viz [4]). Barnes pohlížel na sociální síť jako na soubor bodů, z nichž některé mohou být spojeny čarou. Barnes také čerpal z práce Jacoba Morena [3], který se zabýval kreslením sociogramů, jenž reprezentovaly vztahy mezi dětmi ve třídě.

V druhé polovině 20. století se kromě antropologie a sociologie analýza sociálních sítí rozšířila do mnoha různých disciplín (ekonomie, psychologie, informatika).

Dnes si pod pojmem sociální síť většina představí webové stránky zprostředkující uživatelům možnost stát se členem virtuální komunity sdružující osoby se společnými zájmy nebo pouze prostor pro trávení volného času. První takováto stránka vznikla v roce 2002 pod názvem Friendster (www.friendster.com), kterou po roce následoval MySpace (www.myspace.com) a stal se velmi populárním. V roce 2004 se objevil Facebook (www.facebook.com) původně zacílený pouze na studenty univerzit, který se do dnešní doby několikanásobně rozrostl, a je považován za nejrozšířenější sociální síť vůbec.

Diskuzní fóra lze také považovat za jistou formu sociální sítě. Vazby mezi aktéry na diskuzním fóru nemusí být na první pohled zřejmé a je nutné je podrobit analýze a pokusit se takové vazby nalézt. Data, která jsem extrahoval z mnou zvolených diskuzních fór, jsem podrobil analýze a pokusil se najít vazby a charakteristiky typické pro sociální sítě.

4.3 Analýza sociální sítě

Analýza sociálních síti (social network analysis) je metoda/přístup, jejímž předmětem zájmu je rozbor sociálních vazeb jedinců. Množina těchto vazeb tvoří sociální síť, v rámci které jsou aktéři pojímáni jako uzly (nodes) a vazby, které je spojují, jako hrany (edges). Prostřednictvím matematického vyjádření vlastností vazeb je možné sociální sít analyticky uchopit coby graf a podrobit jí analýze.

K analýze sociálních sítí se nejčastěji využívají specializované počítačové programy, mezi něž patří např. UCINET, Pajek, NetMiner.

Podobný přístup jsem zvolil i pro vytvoření soc. sítě nad diskuzními fóry. Množina uživatelů, kteří se nejméně jednou účastnili diskuze tvoří uzly sítě. Vazbu, která tyto uzly spojuje, jsem si definoval jako reakci příspěvkem na téma diskuzního fóra. Vazba je v tomto případě orientovanou hranou grafu, směřující od uživatele, který zaslal příspěvek, k uživateli, který založil dané téma, do kterého byl příspěvek zaslán.

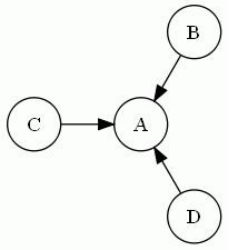
Pokusím se vysvětlit na následujícím, jednoduchém příkladu:

1. Uživatel A založil téma.
2. Uživatelé B, C a D mu na téma odpověděli příspěvkem.
3. Z uzlů B, C a D tedy vede orientovaná hrana do uzlu A.

Viz obrázek (2).

Další kritérium, které jsem zahrnul do analýzy soc. sítě je síla vazby. Jak jsem zmínil výše, účastí v tématu vzniká vazba mezi zakladatelem tématu a přispěvatelem. Nicméně jak v rámci jednoho tématu tak v rámci celého diskuzního fóra se může výše popsaná vazba (mezi stejnými dvěma uživateli) vyskytnout několikrát. Čím větší výskyt vazeb mezi stejnými dvěma uživateli, tím silnější je vazba mezi nimi.

Pro získání všech vazeb, které se v daném diskuzním fóru vyskytují, je potřeba získat z databáze fóra SQL dotazem všechny vazby mezi uživateli, jež založili téma , a uživateli,



Obrázek 2: Demonstrace vazeb (orientovaných hran) v diskuzním fóru

jež do těchto témat přispívali. Výsledkem dotazu je seznam orientovaných hran grafu a síly dané vazby (viz ukázka SQL dotazu (5)).

```
SELECT t.id_user T, p.id_user P, COUNT(*) Pocet FROM post p, topic t
WHERE t.id_topic = p.id_topic AND t.id_user != p.id_user
GROUP BY t.id_user, p.id_user;
```

Výpis 5: Tento SQL dotaz vrací seznam vazeb a jejich sílu.

Freeman [5] píše, že primárním využitím teorie grafů v analýze sociálních sítí je zjistit, jak důležitou roli mají aktéři na individuální nebo skupinové úrovni analýzy. K tomu se používá výpočet tzv. centrality, která popisuje vlastnosti uzlu v rámci celého grafu. Freeman uvádí, že nejpoužívanějšími mírami centralit jsou *degree*, *closeness* a *betweenness*.

V závislosti na tom kolik vazeb/hran z daného uzlu vychází či do něj směřuje lze pro každý uzel spočítat tzv. *degree centrality*, která může blíže specifikovat roli uzlu uvnitř celé sociální sítě (grafu). V praxi může *degree centrality* uzlu vyjadřovat důležitost osoby ve společnosti, důležitost místnosti v budově nebo např. důležitost křižovatky v dopravní síti.

Například Valdis E. Krebs [6] popisuje mapování sociální sítě teroristických buněk po událostech 11. září, 2001. Krebs využívá jak *degree centrality*, tak *closeness* a *betweenness* k analýze vazeb mezi 19 únosci letadel ze zmiňovaného teroristického útoku.

4.3.1 Degree centrality

Knoke [1] popisuje *Degree centrality* u neorientovaných grafů jako součet vazeb, které se s daným uzlem váží. Velmi často je interpretována např. jako riziko, že daný uzel zasáhne cokoliv, co se šíří danou sítí (např. virus nebo informace). U orientovaných grafů je možno rozlišovat vstupní (indegree) a výstupní (outdegree) centralitu. Vstupní centralita je dána počtem vazeb (orientovaných hran) směřujích do uzlu, výstupní centralita je dána počtem vazeb z uzlu vycházejících. V sociálních sítích by se dala vstupní centralita chápat jako určitá míra popularity a výstupní jako družnost, či společenskost (viz [1]).
Výpočet centrality uzlu popisuje dle Knoka [1] následující výraz:

$$C_D(N_i) = \sum_{j=1}^{g} x_{ij}(i \neq j)$$

Kde $C_D(N_i)$ je *degree centrality* uzlu *i* v grafu a $\sum_{j=1}^{g} x_{ij}$ je součet přímých vazeb, kterými se uzel *i* váže s *g-1* jinými (*j*) uzly. $(i \neq j)$ vylučuje vazbu uzlu *i* sama na sebe.

Takto vypočtená *degree centrality* ovšem nereflektuje pouze propojenost uzlů, ale také velikost sítě. Konkrétní *degree centrality* uzlu může znamenat, že je uzel dobře propojen v malé síti, nebo spojen pouze s několik jinými uzly ve velké síti.

Aby eliminovali efekt, jaký má velikost sítě na *degree centrality* uzlů, zavedli Wasserman a Faust (viz [7]) normalizovanou míru centrality:

$$C'_D(N_i) = \frac{C_D(N_i)}{g-1}$$

Normalizovaná *degree centrality* uzlu je podíl *degree centrality* uzlu ku g-1 ostatních uzlů. Hodnoty nabývají hodnot od 0.0 (uzel se neváže s žádným jiným uzlem) do 1.0 (uzel je spojen se všemi ostatními uzly přímo). Takto normalizovaná hodnota umožňuje porovnávat centralitu sítí o různém počtu aktérů.

Na základě výše uvedeného vzorce, jsem počítal *indegree* a *outdegree centrality* jednotlivých uzlů (uživatelů) v databázi. Následující SQL dotaz vrátí seznam uživatelů (id) a jejich vstupní centralitu *(indegree centrality)*:

```
SELECT t.id_user U, COUNT(DISTINCT p.id_user) Indegree
FROM post p, topic t WHERE t.id_topic = p.id_topic  AND t.id_user !=  p.id_user
GROUP BY t.id_user;
```

Výpis 6: Ukázka SQL dotazu pro výpočet indegree centrality.

Výsledné hodnoty byly ověřeny programem UCINET, který spočítá *degree centrality* uzlů na základě vstupní matice (seznam hran). Pro výpočet normalizovaných *degree centrality* jsem použil přímo program UCINET.

Program UCINET jsem také použil pro spočítání celkové centralizace sítě, kterou zavedli Wasserman a Faust [7] jako podíl součtu rozdílů *degree centrality* uzlu s největší *degree centrality*, ku všem ostatním uzlům, a (g-1)(g-2). Celková *degree centralization* nabývá hodnot od 0.0 do 1.0. Čím je *degree centralization* blíže 1.0 tím nerovnoměrněji je rozdělena *degree centrality* uzlů v síti. Extrémním případem je graf - hvězda, kde je jeden uzel spojen se všemi ostatními uzly v grafu, a ostatní uzly jsou spojeny právě pouze s tímto jedním uzlem.

Při vizualizaci sociální sítě jsem kromě *degree centrality*, použil také *Weighted degree centrality*.

4.3.2 Weighted degree centrality

Ačkoliv není *weighted degree centrality* tolik využívána jako *degree centrality*, je vhodné ji v případě diskuzních fór použít, jelikož zohledňuje sílu vazby. Pro výpočet *weighted degree centrality* jsou brány v potaz hodnoty hran, které se s daným uzlem váží. Celková *weighted degree centrality* uzlu je potom součet hodnot všech hran svázaných s uzlem. V případě, že je váha (weight) všech hran rovna jedné, degeneruje *weighted degree centrality* na *degree*

centrality. Následující SQL dotaz vrátí seznam uživatelů (id) a jejich *weighted indegree*:

```
SELECT t.id_user U, COUNT(*) Indegree
FROM post p, topic t WHERE t.id_topic = p.id_topic  AND t.id_user != p.id_user
GROUP BY t.id_user;
```

Výpis 7: Ukázka SQL dotazu pro výpočet weighted degree centrality (indegree).

5 Interpretace a vizualizace výsledků

5.1 Statistika

K vizualizaci následujících statistických údajů jsem použil aplikaci **Calc** ze sady **OpenOffice.org 3.2**. V tabulce (2) jsou souhrnné údaje z databáze obou diskuzních fór.

Diskuzní fórum	**cpukforum.com**	**ironfactor.cz**
Počet uživatelů	2121	801
Počet příspěvků	123799	106264
Počet témat	15758	2800

Tabulka 2: Souhrnné údaje

Následující graf zobrazuje aktivitu uživatelů v průběhu dne. Umožňuje porovnat aktivitu uživatelů na obou diskuzních fórech. Jelikož u každého příspěvku v databázi eviduji čas zaslání, počet příspěvků zaslaných za hodinu jsem získal následujícím SQL dotazem:

```
SELECT count(p.id_post) 'Pocet Prispevku', hour(date) 'Hodina' FROM post p GROUP BY hour(
    date) ORDER BY 2;
```

Výpis 8: Tento SQL dotaz zobrazí počet příspěvků za hodinu.

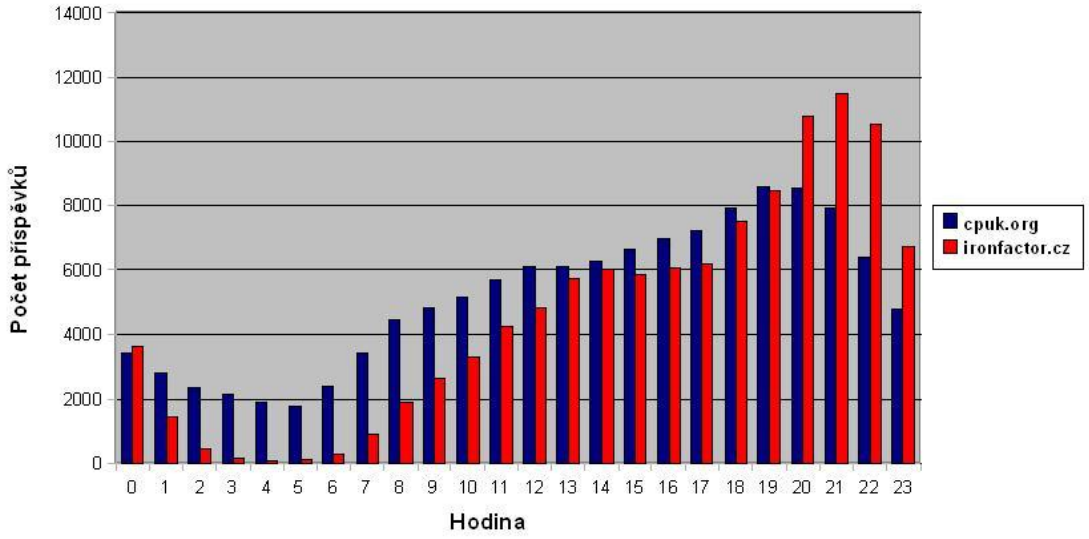
Např. hodina **10** reprezentuje počet příspěvků zaslaných v časovém intervalu 10:00:00 až 10:59:59. Hodina **11** reprezentuje interval 11:00:00 až 11:59:59 atd. V grafu (3) je vidět, že aktivita uživatelů v průběhu dne je víceméně na obou dvou fórech stejná. Kulminuje v pozdních odpoledních hodinách a večer. Naopak útlum nastává v noci a ráno. Je třeba zmínit, že graf fóra cpukforum.com, které je anglické a sdružuje lidi z různých časových pásem, je po celé délce amplifikován pravděpodobně pravě aktivitou uživatelů žijících mimo Evropu. Zvýšená aktivita na fóru ironfactor.cz od 21:00 do půlnoci oproti diskuznímu fóru cpukforum.com. by mohla poukazovat na mladší skupinu uživatelů.

V grafu (4) jsem zobrazil aktivitu (počet příspěvků za den) v průběhu jednoho týdne. K získání údajů posloužil v tomto případě následující SQL dotaz:

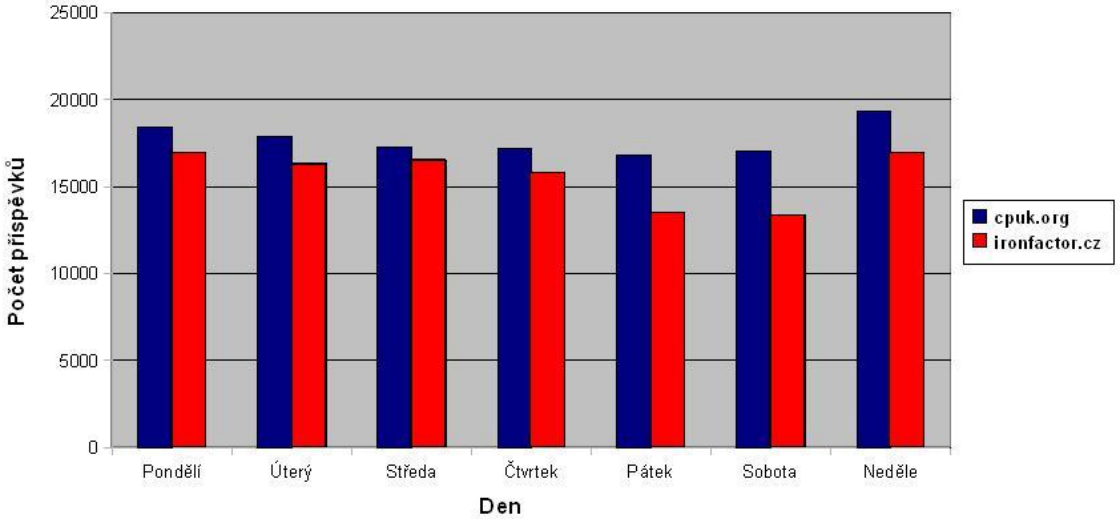
```
SELECT count(p.id_post) 'Pocet Prispevku', dayname(date) 'Den'
FROM post p GROUP BY dayname(date) ORDER BY count(p.id_post) DESC;
```

Výpis 9: Tento SQL dotaz zobrazí počet příspěvků za den v týdnu.

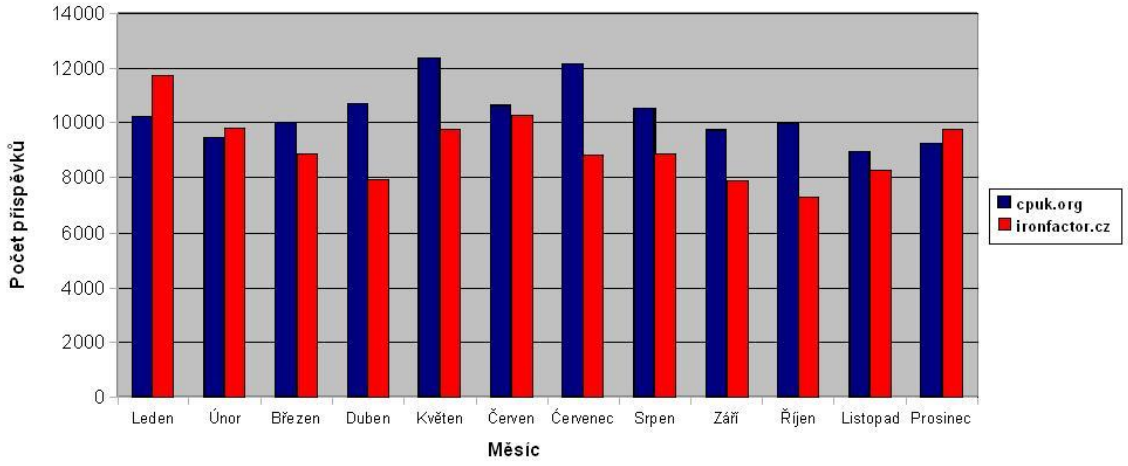
Z grafu je zřejmé, že největší aktivita na obou diskuních fórech nastává na konci (neděle) a na začátku týdne (pondělí). Naopak nejméně aktivní jsou lidé v pátek a v sobotu. Určitě zde hrají velkou roli lidé, kteří v pátek a o víkendu dávají přednost jiným aktivitám a nemají tak přístup k internetu. Rozdíly zde rozhodně nejsou tak viditelné jako v případě denní aktivity a lze říci, že rozdíly v aktivitě během pracovních dnů (pondělí - pátek) jsou minimální.



Obrázek 3: Graf znázorňující počet příspěvků za hodinu v průběhu dne



Obrázek 4: Graf znázorňující počet příspěvků za den v průběhu týdne



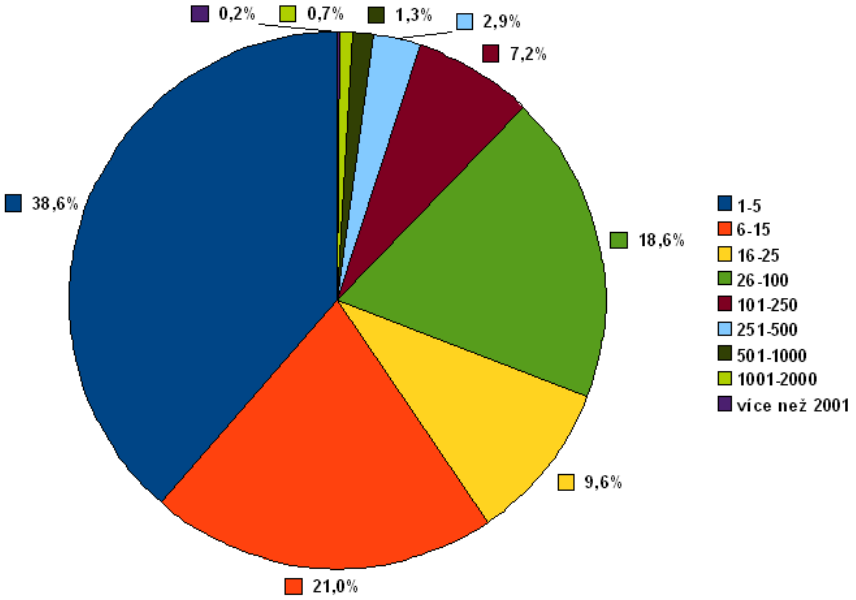
Obrázek 5: Graf znázorňující počet příspěvků za měsíc v průběhu roku

Graf (5) znázorňuje počet příspěvků zaslaných v jednotlivých měsících za celou dobu běhu fóra. Z grafu je zřejmé, že obě fóra vykazují zvýšenou aktivitu s příchodem nového roku. Naopak v průběhu roku je vidět, že fórum cpukforum.com, sdružující pěstitele exotických rostlin a obecně lidi s bližším vztahem k přírodě, převyšuje v měsících duben, květen a obecně letních měsících svou aktivitou fórum ironfactor.cz. Obě diskuzní fóra vykazují pokles aktivity s koncem léta a příchodem podzimu. Je zřejmé, že vliv ročních období ovlivňuje aktivitu uživatelů na diskuzních fórech. Níže uvedený SQL dotaz vrátí počet příspěvků zaslaných za jednotlivé měsíce.

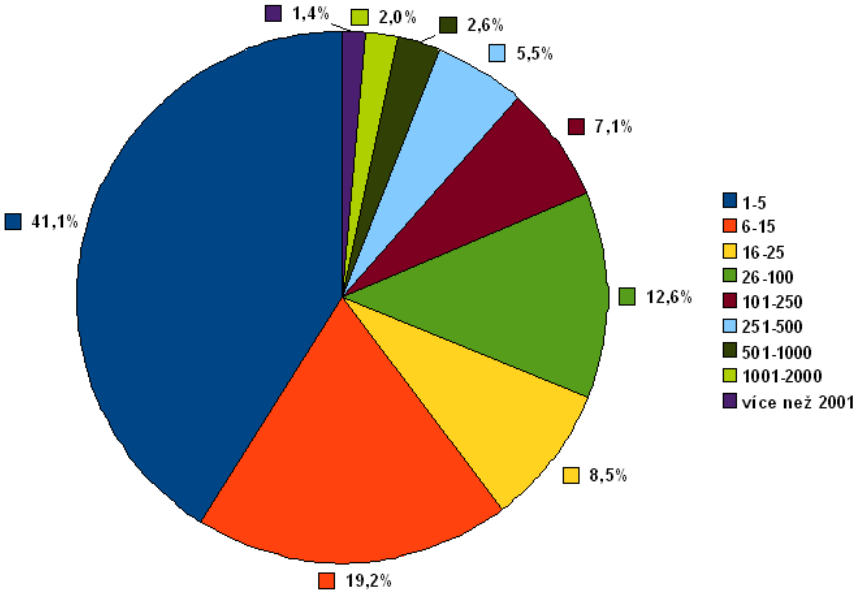
```
SELECT count(p.id_post) 'Pocet Prispevku', monthname(date) 'Mesic'
FROM post p GROUP BY monthname(date) ORDER BY month(date);
```

Výpis 10: Tento SQL dotaz zobrazí počet příspěvků za měsíc v průběhu roku.

V kruhových diagramech (6) a (7) jsem za použití SQL dotazu (11) uživatele jednotlivých diskuzních fór seskupil podle počtu příspěvků do intervalů, abych zjistil zda se fóra liší v poměrech aktivních či neaktivních uživatelů. Na obou diskuzních fórech tvoří přibližně stejný podíl (rozdíl 1-2 procenta) uživatelé s minimálním počtem příspěvků (1-5, 6-15, 16-25 a 26-100). U obou fór tvoří největší podíl uživatelé, kteří zaslali pouze 1-5 příspěvků (40%), což jsou víceméně pasivní uživatelé, kteří svou aktivitou nemohou běh fóra nijak zásadně ovlivnit. Stejně tak s rostoucím počtem příspěvků klesá podíl uživatelů na fóru. Jediný rozdíl, který jsem vypozoroval, se týká uživatelů, kteří zaslali 250 a více příspěvků. Tady je procentuální podíl na fóru ironfactor.cz dvojnásobný. Což značí, že ironfactor.cz má větší množství velmi aktivních uživatelů.



Obrázek 6: Poměr aktivních uživatelů vzhledem k počtu příspěvků na fóru cpukforum.com



Obrázek 7: Poměr aktivních uživatelů vzhledem k počtu příspěvků na fóru ironfactor.cz

```
SELECT '1−5' A, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet BETWEEN
    1 AND 5
UNION ALL
SELECT '6−15' B, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet BETWEEN
    6 AND 15
UNION ALL
SELECT '16−25' C, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet BETWEEN
    16 AND 25
UNION ALL
SELECT '26−100' D, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet BETWEEN
    26 AND 100
UNION ALL
SELECT '101−250' E, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet BETWEEN
    101 AND 250
UNION ALL
SELECT '251−500' F, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet BETWEEN
    251 AND 500
UNION ALL
SELECT '501−1000' G, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet BETWEEN
    501 AND 1000
UNION ALL
SELECT '1001−2000' H, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet BETWEEN
    1001 AND 2000
UNION ALL
SELECT '2001− ' I, COUNT(x.id_user) FROM (
SELECT COUNT(*) Pocet, id_user FROM post p GROUP BY id_user) x WHERE Pocet >= 2001
;
```

Výpis 11: Tento SQL dotaz roztřídí uživatele do intervalů dle jejich celkového počtu příspěvků.

5.2 Vizualizace sociální sítě

Pro kreslení grafů, reprezentujících sociální síť diskuzního fóra jsem použil aplikaci **Graphviz (v 2.26.3)**. Je třeba zmínit, že vstupem pro vykreslení grafu v Graphvizu je textový soubor se specialní syntaxí, popisující seznam uzlů a hran. Graphviz následně z daného txt souboru vykreslí obrázek grafu v požadovaném formátu (např. jpg). Pro korektní vykreslení grafu bylo třeba nejprve SQL dotazem získat tabulku, reprezentující seznam hran (dvojice uzlů), případně sílu hrany. Následně jsem musel v jazyce C♯ napsat konzolovou aplikaci, která data uložená v tabulce přepíše do textového souboru vyhovujícího formátu Graphvizu (viz dokuemntace [11]). Následuje ukázka vstupního formátu:

```
graph G {
node [shape = circle, fixedsize=true, style= filled , label="", fontcolor=red, fontname=arial,
      fontsize=22];
2 [width=0.00412, fillcolor = black]
4 [width=0.00674, fillcolor = black]
8 [width=0.02172, fillcolor = black]
10 [width=0.00412, fillcolor = black]
12 [width=0.00412, fillcolor = black]
33 [width=0.2, fillcolor = black, label=33]
 ...
 ...
2--183[penwidth=0.01]
2--231[penwidth=0.01]
2--629[penwidth=0.01]
2--685[penwidth=0.01]
2--819[penwidth=0.01]
 ...
 ...
}
```

Výpis 12: Ukázka vstupního formátu programu Graphviz.

Jelikož zpracování výše popsaného SQL dotazu (5) trvá i několik minut a je třeba ho provést u obou databází, bylo velice výhodné uložit výsledek složitějšího SQL dotazu do nové tabulky a poté už jej volat pouze jako *SELECT * FROM graph;*.

Pro vizualizaci síly hran v grafu a *degree centrality* uzlů (uživatelů) bylo třeba v přepisovacím programu spočítat sílu vazeb a centralitu uzlů. Najít maximální hodnoty a na základě těchto maxim rozpočítat zbylé hodnoty a přepočítat pro ně tloušťku pera, jakou je má Graphviz vykreslit. Takto zůstává zachován skutečný poměr silných a slabých vazeb (uzlů) v sociální síti. Stejnou metodu jsem použil pro výpočet velikosti jednotlivých uzlů. Velikost uzlu je přímo úměrná jeho vstupní či výstupní centralitě.

Pro grafy znázorňující vazby mezi uživateli a míru centrality platí:

1. Čím větší uzel, tím vetší je jeho míra centrality.
2. Čím silnější čára, tím silnější je vazba mezi dvěma uzly jež spojuje.

Síla vazby je znázorněna pouze v grafech znázorňujících *weighted degree centrality*.

U ostatních grafů je síla vazby rovna jedné a je u všech uzlů stejná. Síla pera kreslící vazby je záměrně snížena, aby vynikly jednotlivé uzly.

Uzel s největší centralitou v grafu je označen červeným popiskem udávající ID uzlu v databázi.

Nepřehlednost či špatná čitelnost je zpravidla zapříčiněna velkou hustotou vazeb a nevhodným vykreslovacím algoritmem.

5.2.1 Degree centrality

Centralizace sítě	
cpukforum.com	37.49%
ironfactor.cz	56.86%

Tabulka 3: Degree centrality: centralizace sítě

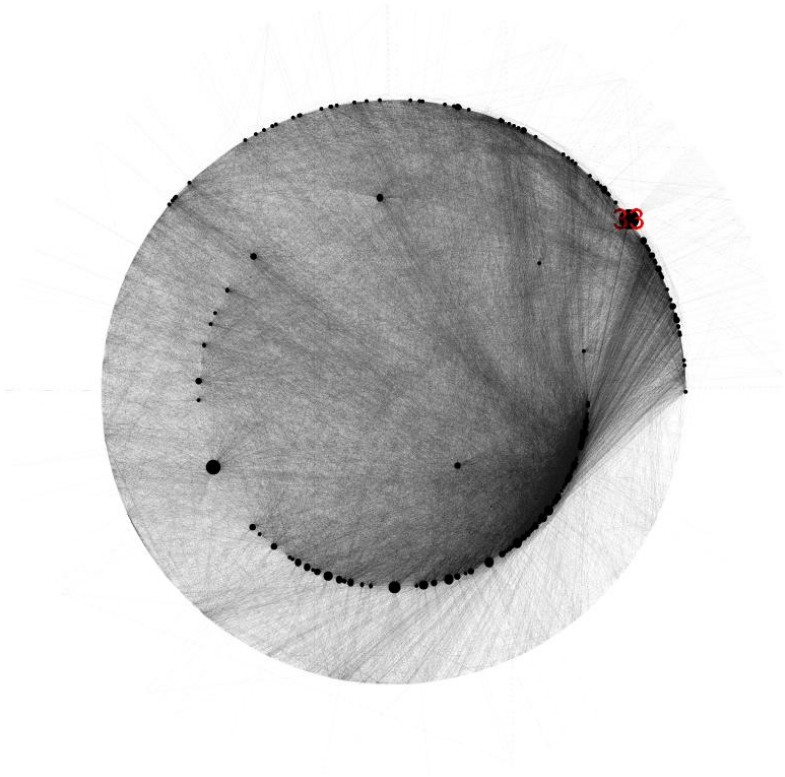
Následující tabulka (4) obsahuje 10 uživatelů (ID) diskuzního fóra cpukforum.com s největší vstupní (Indegree) a výstupní (Outdegree) centralitou a odpovídající normalizovanou centralitou ($C_D^{'}(N_i)$).

Z tabulky (4) vyplývá, že *indegree centrality*, jinými slovy popularita uživatele, nemusí

cpukforum.com					
ID	**Indegree centrality**	$C_D^{'}(N_i)$	**ID**	**Outdegree centrality**	$C_D^{'}(N_i)$
33	534	0.259	819	729	0.353
737	355	0.172	217	580	0.281
217	287	0.139	737	565	0.274
1144	275	0.133	22	552	0.267
627	255	0.123	3160	434	0.210
819	252	0.122	629	337	0.163
1702	245	0.119	356	335	0.162
533	240	0.116	106	301	0.146
1643	226	0.109	2092	301	0.146
356	211	0.102	564	295	0.143

Tabulka 4: Degree centrality: cpukforum.com

nutně znamenat, že uživatel je také velice aktivní (má velkou *outdegree centrality*). Pouze uživatelé 4 uživatelé z 10 se objevují v obou tabulkách, to znamená, že jsou velmi populární a zároveň velmi aktivní. Dále je třeba zmínit, že rozdíl mezi prvním a desátým uživatelem v jak indegree tak outdegree centrality je téměř dvojnásobný. Na uživatele s ID 33 reagovalo téměř o 200 uživatelů více než na druhého uživatele v žebříčku, ID 737. Stejně tak uživatel s ID 819 oslovil o 150 lidí více než uživatel 217 atd. Obrázek (8) pak



Obrázek 8: Indegree centrality, cpukforum.com

reprezentuje sociální síť diskuzního fóra cpukforum.org. Velikost uzlu je přímo úměrná jeho *indegree centrality* a uživatel s největší vstupní centralitou (ID 33) je označen červeně.

Následující tabulka (5) obsahuje 10 uživatelů (ID) diskuzního fóra ironfactor.cz s největší vstupní (Indegree) a výstupní (Outdegree) centralitou a odpovídající normalizovanou centralitou ($C'_D(N_i)$).
Z tabulky (5) i grafu (9) je zřejmé, že uživatel s ID 3 má několikrát větší vstupní centralitu než ostatní uživatelé fóra ironfactor.cz a hraje v celé sociální síti důležitou roli. Reagovala na něj téměř polovina (396) z celkového počtu (800) uživatelů. Z obrázku (9) je jasné, že síť fóra ironfactor.cz je oproti cpukforum.com centralizovaná právě kolem jednoho až dvou uživatelů a role ostatních uživatelů v síti jsou v porovnání zanedbatelné.

5.2.2 Weighted degree centrality

Následující tabulka obsahuje 10 uživatelů (ID) diskuzního fóra cpukforum.com s největší weighted indegree a weighted outdegree centrality.
Tabulka (6) weighted degree centrality zohledňuje nejen propojenost uzlů, ale také sílu vazby. Jinými slovy je zde brána v úvahu nejen popularita, ale také komunikační schop-

ironfactor.cz					
ID	**Indegree centrality**	$C_D^{'}(N_i)$	**ID**	**Outdegree centrality**	$C_D^{'}(N_i)$
3	396	0.503	156	360	0.457
156	221		125	285	0.362
17	142	0.180	49	250	0.317
4	138	0.175	137	246	0.312
65	130	0.165	3	245	0.311
343	122	0.155	123	241	0.306
226	121	0.154	37	229	0.291
55	120	0.152	65	220	0.279
191	117	0.148	400	210	0.266
422	117	0.148	226	204	0.259

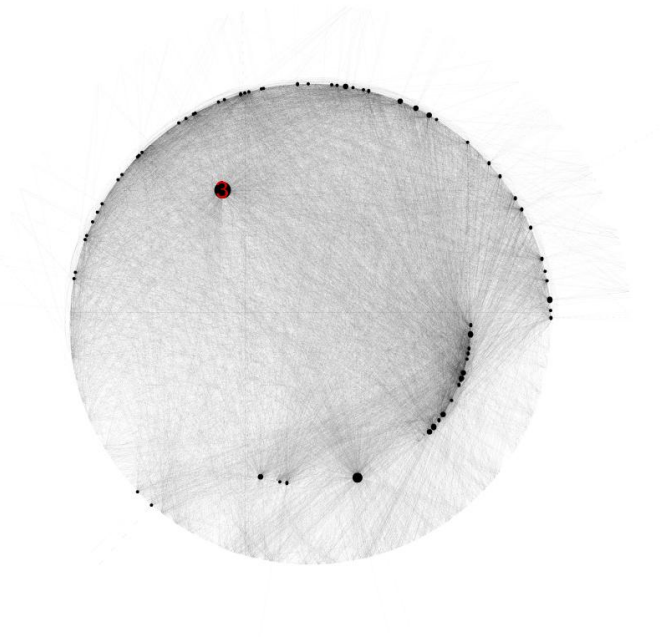
Tabulka 5: Degree centrality: ironfactor.cz

cpukforum.com			
ID	**Weighted indegree centrality**	**ID**	**Weighted outdegree centrality**
737	1473	819	2640
1144	1186	217	2159
33	1129	22	2064
1643	988	3160	1731
627	946	629	1561
819	944	737	1450
217	939	356	1035
1702	850	328	1033
114	831	1702	1024
827	824	2092	962

Tabulka 6: Weighted degree centrality: cpukforum.com

nosti. Vysoká weighted outdegree neznamená, že dotyčný pouze navazoval kontakty, ale také aktivně komunikoval a posiloval vazby. Na uživatele s ID 33 sice reagovalo nejvíce lidí, ale pokud se veme v úvahu síla vazby, je až na třetím místě. A naopak na uživatele 737 reagovalo o 150 lidí méně, nicméně součet vah vazeb z něj dělá velmi důležitou osobu v celé sociální síti.

Je přirozené, že velký počet unikátních vazeb, bude zároveň implikovat i větší weighted degree centrality. Více lidí vyprodukuje více silnějších vazeb. Obrázek (10) zobrazuje Weighted Outdegree Centrality fóra cpukforum.com a sílu vazeb. Je evidentní, že na fóru je obravské množství jak slabých, tak několik velmi silných vazeb mezi významnými uživateli. Oproti fóru ironfactor.cz je zde očividně větší základna signifikatních členů, jejichž



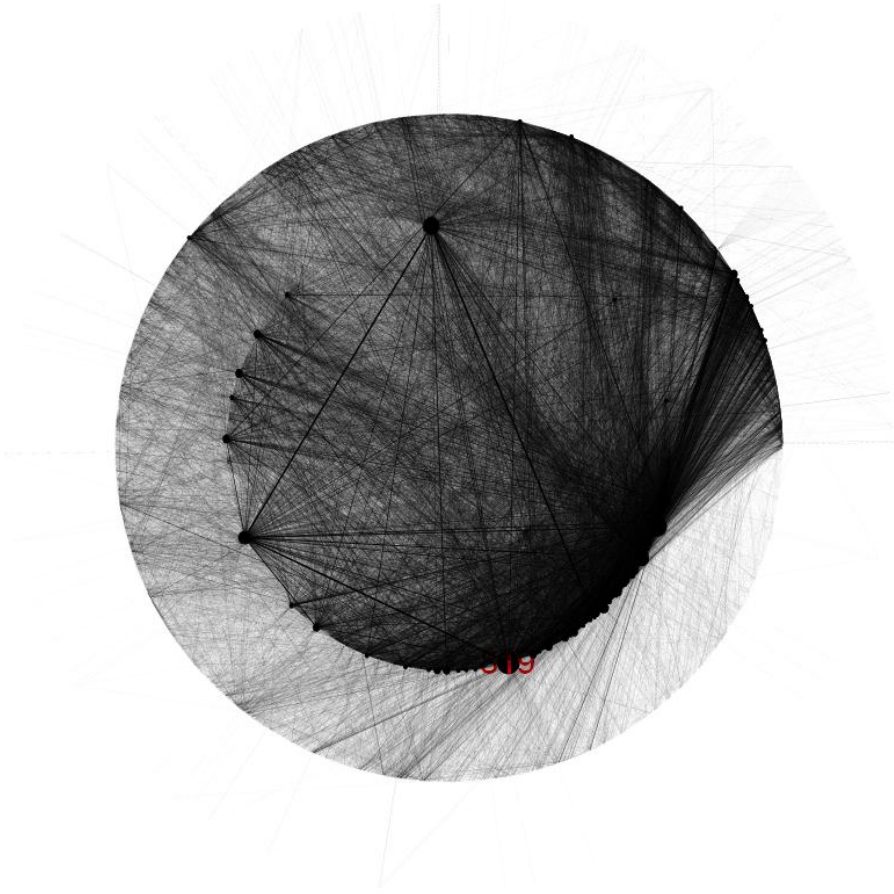
Obrázek 9: Indegree centrality, ironfactor.cz

množina tvoří základ celé sociální sítě, oproti síti silně centralizované kolem jednoho uživatele na fóru ironfactor.cz.

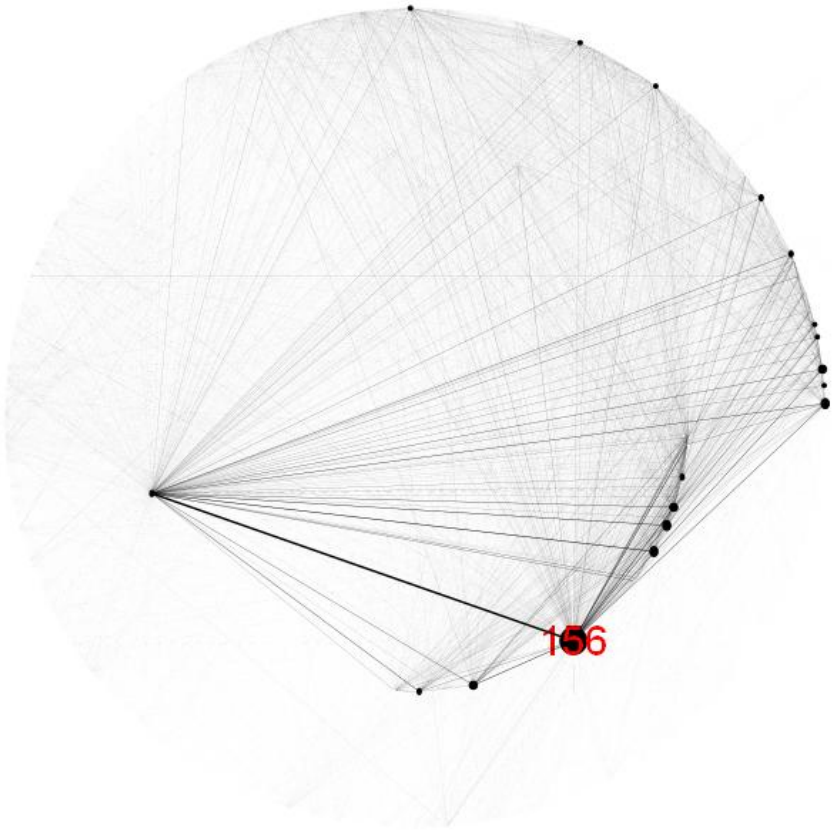
Tabulka (7) obsahuje 10 uživatelů (ID) diskuzního fóra ironfactor.cz s největší weighted indegree a weighted outdegree centrality.

ironfactor.cz			
ID	**Weighted indegree centrality**	**ID**	**Weighted outdegree centrality**
3	14021	156	8547
156	6600	49	3164
4	3173	125	2847
226	2497	137	2826
17	1883	65	2268
65	1755	226	2218
191	1644	320	2091
343	1284	123	1953
24	1211	37	1789
55	1190	3	1700

Tabulka 7: Weighted degree centrality: ironfactor.cz



Obrázek 10: Weighted outdegree centrality, cpukforum.com



Obrázek 11: Weighted outdegree centrality, ironfactor.cz

Tabulka 7 pouze potvrzuje, že středobodem celé sociální sítě fóra ironfactor.cz je uživatel s ID 3, na kterého směřovalo celkem 14021 reakcí. Naopak jeho weighted outdgree je velmi nízká, což svědčí o jeho ne příliš velké družnosti. Uživatel 156, jehož weighted indegree je sice pouze poloviční oproti uživateli 3, má však nejvíce výstupních reakcí a je tedy nejaktivnějším uživatelem fóra.

6 Závěr

Cílem mé bakalářské práce bylo extrahovat vybranná data z Webu, v mém případě z diskuzních fór, a pokusit se nad těmito daty vytvořit sociální síť a tu analyzovat.

Data z diskuzních fór se podařilo úspěšně extrahovat a celý proces se několikanásobně urychlil paralelním zpracováváním několika webových stránek současně za použití vláken. Oddělení části programu, která stahuje data z internetu, a části, které data zpracovává a ukládá, zvýšilo efektivitu celého procesu.

Zvolený způsob ukládání dat, do vhodně navržené databáze, se ukázal být velice efektivní a umožnil pohodlný přístup a zpracovávání většího množství dat (řádově desetitisíce položek).

Takto extrahová data jsem zpracoval a za využití vazeb mezi účastníky diskuzního fóra jsem nad těmito vazbami sestrojil graf reprezentující sociální síť. Takto vytvořenou síť jsem poté analyzoval.

Z hlediska analýzy sociálních sítí bylo velmi zajimavé pozorovat společné i rozdílné rysy uživatelů tvořících danou sociální síť, zejména pak jejich aktivitu v různých časových horizontech. Velkou pozornost jsem věnoval zvláště zjišťování důležitosti uživatele v soc. síti, reprezentované mírou centrality (degree centrality). Bylo tak možné určit, zda je uživatel mezi ostatními populární či komunikačně zdatný.

Díky zvolené vizualizaci sociální sítě, se získané výsledky staly přehlednými a z obrázků a grafů reprezentujících soc. síť bylo možno alespoň částečně interpretovat role jednotlivých uživatelů uvnitř soc. sítě a sílů vztahů mezi nimi.

Závěrem by bylo vhodné zmínit, že z analytického hlediska je rozbor sociálních síti nejenom časově, ale i technicky náročná disciplína, a opírá se o obsáhlé teoretické studie. Zvolená data by bylo možné podrobit dalšímu studiu za pomoci jiných, zde nepopsaných analytických metod (betweenness, closeness apod.), jehož zpracování by však mnohokrát prekročilo rozsah této bakalářské práce.

Petr Dolenský

7 Reference

[1] KNOKE, David, YANG, Song, *Social Network Analysis, 2nd Edition*, India: Sage Publications, 2008, ISBN: 978-1-4129-2749-9.

[2] NAGEL, Christian, *C♯, Programujeme profesionálně*, Brno: Computer Press, a.s., 2009, ISBN: 978-80-251-2401-7.

[3] MORENO, J.L., *Who shall survive?*, Washington, DC: Nervous and Mental Disease Publishing Company, 1934.

[4] BARNES, J., *Class and committees in a Norweigian island parish, Human Relations, 7*, England, 1954.

[5] FREEMAN, L.C., *Centrality in social networks: I. Conceptual clarification. Social networks, 1 (3)*, Elsevier Sequoia S.A., Lausanne, 1979.

[6] KREBS E. Valdis, *Mapping Networks of Terrorist Cells, Connections 24 (3)*, orgnet.com, INSNA, 2002.

[7] WASSERMAN, S., FAUST, K., *Social network analysis: Methods and applications*, New York: Cambridge University Press, 1994.

[8] http://msdn.microsoft.com/cs-cz/library/default.aspx, *Microsoft Developer Network*.

[9] http://en.wikipedia.org/wiki/Social_network, *Social network, Wikipedia.org*.

[10] http://www.faculty.ucr.edu/hanneman/nettext/C10_Centrality.html, *Web University of California, Riverside*.

[11] http://www.graphviz.org/Documentation.php, *Dokumentace programu Graphviz*.

[12] http://en.wikipedia.org/wiki/Internet_forum, *Internet forum, Wikipedia.org*.

[13] http://dev.mysql.com/doc/, *MySQL dokumentace*.